РОССИЙСКАЯ АКАДЕМИЯ НАУК

СИБИРСКОЕ ОТДЕЛЕНИЕ

А В Т О М Е Т Р И Я

2008, том 44, № 5

УДК 004.82

ПАРАМЕТРИЧЕСКАЯ ИДЕНТИФИКАЦИЯ НЕЧЕТКИХ МОДЕЛЕЙ НА ОСНОВЕ ГИБРИДНОГО АЛГОРИТМА МУРАВЬИНОЙ КОЛОНИИ*

И. А. Ходашинский, П. А. Дудин

Томский государственный университет систем управления и радиоэлектроники, г. Томск
E-mail: hodashn@rambler.ru

Рассмотрено применение алгоритма муравьиной колонии для решения задачи параметрической идентификации нечеткой модели. Определен переход от непрерывной оптимизации к дискретной посредством построения полного ориентированного графа поиска решения. Рассмотрен градиентный алгоритм в качестве второго этапа оптимизации. Проведен эксперимент по исследованию работы алгоритмов оптимизации и нечеткой системы.

Введение. Нечеткие системы находят широкое применение в таких проблемных областях, как извлечение знаний и проектирование регуляторов, прогнозирование и моделирование, управление и принятие решений. Однако используются эти системы без должного обоснования выбора структуры и параметров модели. Указанные задачи должны быть решены на этапе идентификации модели. Наиболее часто используемые подходы к идентификации параметров нечеткой модели – генетические алгоритмы и нейронные сети [1–5]. В работе [6] предложен для этих целей алгоритм муравьиной колонии (АМК). Однако применение данного метода ограничивает точность его работы, повышение которой возможно путем комбинированного использования алгоритма муравьиной колонии и градиентного метода (ГМ). Идея совместного использования различных техник не нова, на этом принципе сформулировано направление, получившее название «мягкие вычисления» [7], но идентификация нечетких моделей на основе АМК и ГМ описывается впервые.

Целью предлагаемой работы является описание возможности совместного применения АМК и ГМ для идентификации нечетких моделей.

Постановка задачи. Нечеткая модель может быть сформирована либо на основе знаний эксперта, либо на основе наблюдаемых данных, либо на совместном использовании знаний и данных. Нечеткая система, построенная

* Работа выполнена при поддержке Российского фонда фундаментальных исследований (проект № 06-08-00248).

на основе нечеткой модели, выступает в качестве универсального аппроксиматора. Создание аппроксимирующего описания объекта разделяется на следующие этапы: подбор структуры модели, оценивание параметров модели, выбор критерия качества аппроксимации и метода оптимизации выбранного критерия.

В данной работе рассматривается нечеткая система типа синглтон, в которой n входных переменных, m нечетких правил, каждое из которых имеет следующий вид:

$$\text{правило } j\text{: ЕСЛИ } x_1 = A_{1j} \text{ И } x_2 = A_{2j} \text{ И} \ldots \text{И } x_n = A_{nj} \text{ ТО } r_j,$$

где r_j – действительное число, $r_j \in \Re$. Нечеткая система осуществляет отображение F: $\Re^n \rightarrow \Re$, заменяя оператор нечеткой конъюнкции произведением, а оператор агрегации нечетких правил – сложением. Тогда выходное значение $F(\mathbf{x})$ вычисляется следующим образом:

$$F(\mathbf{x}) = \frac{\sum_{j=1}^{m} r_j \prod_{i=1}^{n} \mu_{A_{ij}}(x_i)}{\sum_{j=1}^{m} \prod_{i=1}^{n} \mu_{A_{ij}}(x_i)},$$

где $\mathbf{x} = [x_1, \ldots, x_n]^T \in \Re^n$; $\mu_{A_{ij}}(x_i)$ – функция принадлежности (ФП) нечеткого понятия A_{ij}.

Известно несколько типов ФП: треугольные, трапециевидные, гауссовы, параболические, каждая из которых задается определенным набором параметров. Без потери общности в предлагаемой работе рассматриваются треугольные функции принадлежности, для определения которых необходимо три параметра.

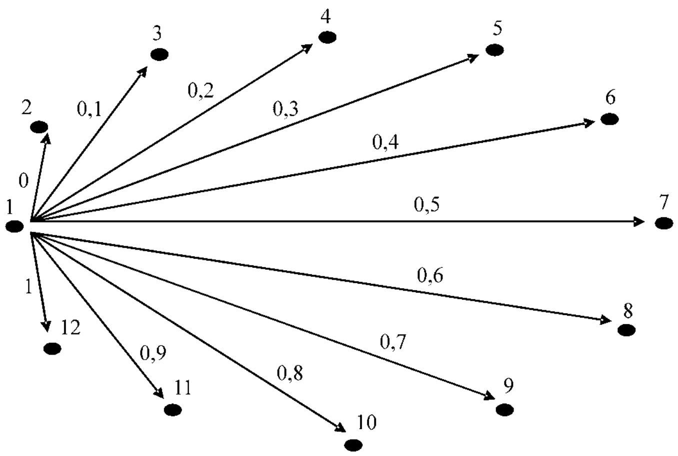
Основой для оценивания параметров функций принадлежности является таблица наблюдений либо тестовая функция $f(\mathbf{x})$.

Проблема идентификации нечетких моделей решается как задача оптимизации, цель которой – найти параметры модели так, чтобы одна из ошибок

$$\frac{\sum_{i}^{N} \left| f(\mathbf{x}_i) - F(\mathbf{x}_i) \right|}{N}, \quad \frac{\sqrt{\sum_{i}^{N} (f(\mathbf{x}_i) - F(\mathbf{x}_i))^2}}{N} \quad \text{или} \quad \max_i (|f(\mathbf{x}_i) - F(\mathbf{x}_i)|)$$

между тестовыми $f(\mathbf{x})$ и модельными $F(\mathbf{x})$ значениями могла быть минимизирована. Тогда проблема сводится к проведению поиска в многомерном пространстве, координаты которого соответствуют параметрам нечеткой модели. В силу того что поверхность для поиска в указанном пространстве имеет сложный рельеф, применение только лишь ГМ здесь неэффективно. Для решения данной проблемы на первом этапе поиска предлагается использовать такую метаэвристику, как алгоритм муравьиной колонии, а на втором этапе – градиентный метод.

Алгоритм муравьиной колонии. Минимизация ошибки в предлагаемой работе ведется на основе АМК. Идея алгоритма основана на способности муравьев находить кратчайший путь между источником пищи и муравейником без использования визуальной информации [8]. Способность эта об-



Фрагмент начального графа поиска решения для муравьев, начинающих движение из узла 1

условлена выделением муравьями во время движения фермента, который служит маркером.

Каждый муравей предпочитает следовать некоторым указателям, в качестве которых выступает фермент. Со временем фермент улетучивается и естественно меньшее его количество находится на менее популярных маршрутах.

Применение АМК для решения конкретной задачи требует:

1) представления решаемой задачи в виде графа,

2) определения целевой функции,

3) задания эвристического предпочтения одного решения другому.

АМК – процедура дискретной оптимизации, в то время как параметры ФП меняются непрерывно. Переход от непрерывной оптимизации к дискретной осуществляется посредством построения полного ориентированного графа поиска решения, количество вершин в котором определяется точностью нахождения значений параметров. Из каждой вершины выходят дуги с равномерно распределенными значениями нормированных параметров функций принадлежности входных переменных (см. рисунок). Во все вершины графа равномерно распределяются муравьи. Цель муравья в задаче идентификации параметров нечеткой системы – посетить столько вершин, сколькими параметрами задается ФП (например, треугольная – тремя вершинами). Пометки дуг, по которым прошел муравей, будут являться найденным им решением.

Каждая лингвистическая переменная описана несколькими ФП. Муравьи в алгоритме делятся на колонии. Каждая колония муравьев отвечает за нахождение параметров своей ФП.

Количество фермента, наносимого на дуги, пропорционально качеству решения: чем меньше ошибка вывода нечеткой системы, выполненного на выбранных параметрах, тем больше фермента наносится на дуги:

$$\Delta\tau_{ij}^{k}(t) = \begin{cases} Q/L^{k}(t), & \text{если муравей проходил дугу } (i,j), \\ 0 & \text{иначе,} \end{cases}$$

где Q – константа, определяющая количество фермента у отдельного муравья; $L^k(t)$ – значение ошибки на t-й итерации для параметров, выбранных k-м муравьем.

Вероятность выбора муравьем дуги (i, j) на t-й итерации, если дуга не является третьим выбранным параметром ФП, вычисляется по формуле

$$p_{ij}(t) = \frac{\tau_{ij}(t)^{\alpha}}{\sum_{l=1}^{N} \tau_{il}(t)^{\alpha}}, \tag{1}$$

а вероятность выбора k-м муравьем дуги (i, j) на t-й итерации, если дуга является третьим выбранным параметром ФП, находится по формуле

$$p_{ij}^{k}(t) = \frac{\tau_{ij}(t)^{\alpha} \left(\frac{1}{L_j^k(t)} \right)^{\beta}}{\sum_{l=1}^{N} \left(\tau_{il}(t)^{\alpha} \left(\frac{1}{L_l^k(t)} \right)^{\beta} \right)}, \tag{2}$$

где α – параметр, учитывающий важность фермента на пути; β – параметр, учитывающий важность ошибки; $\tau_{ij}(t)$ – интенсивность фермента на дуге между узлами i и j на t-й итерации; $L_o^k(t)$ – значение ошибки на t-й итерации для параметров, выбранных k-м муравьем, если дуга (i, o) является последней, т. е. определяет третий параметр; N – множество узлов, смежных вершине i, причем узел j может быть выбран многократно; в отличие от классического использования алгоритма в нашем случае не формируется запрещенный список узлов; $l \neq i$.

Увеличение количества фермента определяется следующим образом:

$$\tau_{ij}(t+1) = \sum_{k=1}^{M} (\Delta\tau_{ij}^{k}(t) + \tau_{ij}(t))\rho, \tag{3}$$

а испарение фермента – по формуле

$$\tau_{ij}(t+1) = \tau_{ij}(t)(1-\rho). \tag{4}$$

Здесь $\rho \in [0; 1]$ – коэффициент снижения интенсивности фермента; M – количество муравьев, прошедших по дуге (i, j).

Приведем алгоритм оптимизации параметров ФП.

Шаг 1. Задать начальные параметры алгоритма и нечеткой системы.

Шаг 2. Задать популяции муравьев в колониях.

Шаг 3. Для всех муравьев текущей колонии определить три дуги, используя 2 раза формулу (1) и затем формулу (2).

Шаг 4. Передать в нечеткую систему значения параметров ФП, определенных муравьями текущей колонии, и вычислить ошибки. Если параметры, переданные муравьем в нечеткую систему, лучше текущих, то сохранить новые значения параметров.

Шаг 5. Если имеется следующая колония, то сделать ее текущей и перейти на шаг 3, иначе перейти на шаг 6.

Шаг 6. Вычислить количество фермента на каждой дуге по формуле (3).

Шаг 7. Вычислить количество испаренного фермента по формуле (4).

Шаг 8. Если условие окончания работы алгоритма выполнено, то выход, иначе перейти на шаг 2.

Условием окончания работы алгоритма является достижение заданного числа итераций либо получение ошибки меньше заданной.

Увеличить точность вычисления параметров можно несколькими путями.

Первое самое очевидное решение – это повышение точности за счет увеличения количества вершин орграфа. Например, для того чтобы точность была 0,01, можно создать граф, у которого 102 вершины. Но увеличение размерности графа на порядок приводит к росту времени вычисления на два порядка.

Для устранения этой проблемы предлагается другой способ – поэтапное увеличение точности. Сначала алгоритм работает с точностью 0,1 и находит оптимальные значения тройки параметров. Затем определенные параметры ограничиваются слева и справа значением 0,1 и в этих пределах берутся значения с шагом 0,01 в качестве пометок нового орграфа. Так процесс продолжается далее в зависимости от требуемой точности. Это позволяет существенно уменьшить время вычислений.

Еще одним способом увеличения точности определения параметров ФП является применение ГМ на втором этапе оптимизации после завершения работы алгоритма с заданной точностью вычисления.

Градиентный метод. Выполнение градиентного алгоритма начинается с некоторого заданного множества модельных параметров. На каждой итерации t вычисляется градиент $\nabla W(P \mid t)$, где $W(P \mid t)$ – целевая функция, которая задает качество сгенерированного решения в зависимости от множества параметров модели P на итерации t. Значение параметра p на итерации $t+1$ определяется как

$$p(t+1) = p(t) + \alpha \nabla W(P \mid t)\big|_p,$$

где α – константа обучения.

Для нечетких моделей типа синглтон с двумя входными переменными, разделенными на пять лингвистических термов, рассмотрим градиентный алгоритм обучения [9]. В обучающих примерах, заданных в виде троек (x_1^*, x_2^*, y^*), для минимизации погрешности решений предложена целевая функция

$$E = \frac{1}{2}(y^* - y)^2,$$

где y^* – желательное выходное значение; y – выходное значение, полученное в результате нечеткого вывода.

Для треугольных функций принадлежности

$$\mu(x)=\begin{cases}(x-a)/(b-a), & a\le x\le b,\\ (x-c)/(b-c), & b\le x\le c,\\ 0 & \text{в других случаях}\end{cases}$$

частные производные

$$\frac{\partial\mu_{1i}}{\partial b_{1i}}=\begin{cases}\dfrac{a_{1i}-x_1}{(b_{1i}-a_{1i})^2},\\ \dfrac{c_{1i}-x_1}{(b_{1i}-c_{1i})^2},\end{cases}\qquad \frac{\partial\mu_{2i}}{\partial b_{2i}}=\begin{cases}\dfrac{a_{2i}-x_2}{(b_{2i}-a_{2i})^2},\\ \dfrac{c_{2i}-x_2}{(b_{2i}-c_{2i})^2},\end{cases}$$

тогда значения параметров на итерации $t+1$ определяются следующим образом:

$$a_{1i}(t+1)=a_{1i}(t)+\frac{\alpha\,(y^*-y)\sum\limits_{j=1}^{5}(r_{ij}-y)\mu_{2j}(x_2^*)}{\sum\limits_{j=0}^{24}\mu_{1l}(x_1^*)\mu_{2m}(x_2^*)}\,\frac{(x_1^*-b_{1i})}{(b_{1i}-a_{1i})^2},$$

$$b_{1i}(t+1)=b_{1i}(t)+\frac{\beta\,(y^*-y)\sum\limits_{j=1}^{5}(r_{ij}-y)\mu_{2j}(x_2^*)}{\sum\limits_{j=0}^{24}\mu_{1l}(x_1^*)\mu_{2m}(x_2^*)}\,\frac{\partial\mu_{1i}(x_1^*)}{\partial b_{1i}},$$

$$c_{1i}(t+1)=c_{1i}(t)+\frac{\lambda\,(y^*-y)\sum\limits_{j=1}^{5}(r_{ij}-y)\mu_{2j}(x_2^*)}{\sum\limits_{j=0}^{24}\mu_{1l}(x_1^*)\mu_{2m}(x_2^*)}\,\frac{(x_1^*-b_{1i})}{(b_{1i}-c_{1i})^2},$$

$$a_{2i}(t+1)=a_{2i}(t)+\frac{\alpha\,(y^*-y)\sum\limits_{j=1}^{5}(r_{ji}-y)\mu_{1j}(x_1^*)}{\sum\limits_{j=0}^{24}\mu_{1l}(x_1^*)\mu_{2m}(x_2^*)}\,\frac{(x_2^*-b_{2i})}{(b_{2i}-a_{2i})^2},$$

$$b_{2i}(t+1)=b_{2i}(t)+\frac{\beta\,(y^*-y)\sum\limits_{j=1}^{5}(r_{ji}-y)\mu_{1j}(x_1^*)}{\sum\limits_{j=0}^{24}\mu_{1l}(x_1^*)\mu_{2m}(x_2^*)}\,\frac{\partial\mu_{2i}(x_2^*)}{\partial b_{2i}},$$

$$c_{2i}(t+1)=c_{2i}(t)+\frac{\lambda(y^*-y)\sum_{j=1}^{5}(r_{ji}-y)\mu_{1j}(x_1^*)}{\sum_{j=0}^{24}\mu_{1l}(x_1^*)\mu_{2m}(x_2^*)}\frac{(x_2^*-b_{2i})}{(b_{2i}-c_{2i})^2},$$

где $i=1-5$; $l=(j\mathrm{mod}5)+1$; $m=(j\mathrm{div}5)+1$; α, β, λ – константы обучения; t – итерация обучения.

Значение r_i консеквента правила на $t+1$-м шаге определяется следующим образом:

$$r_{i+1}(t+1)=r_{i+1}(t)-\gamma\frac{\partial E}{\partial r_{i+1}(t)}=r_{i+1}(t)+\frac{\gamma(y^*-y)\mu_{1k}(x_1^*)\mu_{2n}(x_2^*)}{\sum_{j=0}^{24}\mu_{1l}(x_1^*)\mu_{2m}(x_2^*)},$$

где γ – константы обучения; $i=0-24$; $l=(j\mathrm{mod}5)+1$; $m=(j\mathrm{div}5)+1$; $k=$ $=(i\mathrm{mod}5)+1$; $n=(i\mathrm{div}5)+1$.

Эксперимент. Исследования алгоритма проводились на двух тестовых функциях:

$$f(x_1,x_2)=x_1\sin(x_2),$$

$$f(x_1,x_2)=\sin(2x_1/\pi)\cdot\sin(2x_2/\pi).$$

В эксперименте использовались АМК, нечеткая система, градиентный алгоритм. Цель эксперимента – исследование работы АМК и нечеткой системы при изменении их параметров. Основные параметры по умолчанию для каждого участника эксперимента приведены в табл. 1.

Таблица 1

Исходные параметры проведения эксперимента

Участники эксперимента	Параметры
Алгоритм муравьиной колонии	Количество итераций 100
	Точность 0,1
	Муравьи/узлы 2
	α = 1,0
	β = 0,5
	ρ = 0,7
	Q = 0,1
Нечеткая система	Количество термов 5
Градиентный алгоритм	Коэффициент обучения 0,1

Т а б л и ц а 2

Результаты эксперимента для меняющегося числа итераций

Тестовая функция	Итерации	Усредненная ошибка		Относительное время
		абсолютная	максимальная	
1	50	0,002845726	0,024205667	3,200
	100	0,002721836	0,021214896	5,875
	500	0,002232811	0,017698776	20,500
2	50	0,034109159	0,119509524	4,500
	100	0,031789131	0,116961708	8,857
	500	0,029273961	0,114740514	29,714

В результате проведенных исследований было установлено, что увеличение количества итераций работы алгоритма (табл. 2), количества стартующих из одной вершины муравьев (табл. 3) и значения параметра β (табл. 4) уменьшает ошибку, увеличивая при этом время работы алгоритма. Однако достичь существенного уменьшения ошибки с увеличением значений указанных параметров не удается, особенно этот факт хорошо демонстрируют данные, полученные для второй тестовой функции.

Особо следует отметить эксперимент с изменением значения параметра точности (табл. 5). Увеличение данного параметра на порядок приводит к

Т а б л и ц а 3

Результаты эксперимента для меняющегося соотношения муравьи/вершины

Тестовая функция	Муравьи/ вершины	Усредненная ошибка		Относительное время
		абсолютная	максимальная	
1	2	0,002721836	0,021214896	5,875
	3	0,001864481	0,015989639	6,500
	4	0,001838337	0,015949413	9,500
2	2	0,031789131	0,116961708	8,857
	3	0,030023944	0,110194509	12,600
	4	0,029829003	0,110181491	17,600

Таблица 4

Результаты эксперимента для меняющегося параметра β

Тестовая функция	β	Усредненная ошибка		Относительное время
		абсолютная	максимальная	
1	0	0,002772030	0,022750964	5,400
	0,5	0,002721836	0,021214896	5,875
	1,0	0,001838335	0,015949413	6,250
2	0	0,035395795	0,135066593	8,250
	0,5	0,031789131	0,116961708	8,857
	1,0	0,028940055	0,112478177	9,330

уменьшению ошибки на порядок при работе с первой тестовой функцией. А вот для второй тестовой функции изменения ошибок несущественные.

Такие параметры АМК, как α, Q, ρ, не оказывают значительного влияния на время работы алгоритма и незначительно меняют ошибку вывода (табл. 6). Указанные параметры нуждаются в настройке при работе с конкретной таблицей наблюдений.

Увеличение числа лингвистических термов входных переменных для плавно меняющихся значений не приводит к существенному уменьшению ошибки (табл. 7, функция 1), а при аппроксимации поверхностей со многими

Таблица 5

Результаты эксперимента для меняющегося параметра точности

Тестовая функция	Точность	Усредненная ошибка		Относительное время
		абсолютная	максимальная	
1	0,1	0,002721836	0,021214896	5,875
	0,01	0,000573382	0,004513032	40,250
	0,001	$6{,}80 \cdot 10^{-5}$	0,000505577	64,333
2	0,1	0,031789131	0,116961708	8,857
	0,01	0,016777285	0,097945086	52,000
	0,001	0,016176362	0,097800831	103,000

Т а б л и ц а 6

Результаты эксперимента для меняющегося параметра ρ

Тестовая функция	ρ	Усредненная ошибка		Относительное время
		абсолютная	максимальная	
1	0,4	0,002371453	0,017076440	5,000
	0,7	0,002721836	0,021214896	5,875
	0,9	0,001999847	0,017949413	6,200
2	0,4	0,032322169	0,128145903	9,000
	0,7	0,031789131	0,116961708	8,857
	0,9	0,031949612	0,122880479	9,300

максимумами и минимумами позволяет уменьшить ошибки на порядок, здесь важным является подбор числа термов (см. табл. 7, функция 2).

АМК находит околооптимальные значения параметров нечеткой системы. Более тонкая настройка проводилась с помощью градиентного метода после того, как отработал АМК (табл. 8).

Результаты эксперимента позволяют сделать вывод о том, что для нечеткой системы, аппроксимирующей поверхность без резких изменений, можно найти такие параметры, при которых ошибки появляются в пятом знаке и далее.

Т а б л и ц а 7

Результаты эксперимента для меняющегося числа термов

Тестовая функция	Количество термов	Усредненная ошибка		Относительное время
		абсолютная	максимальная	
1	5	0,002721836	0,021214896	5,875
	7	0,002261576	0,012952662	9,330
	11	0,002198199	0,011147342	15,000
2	5	0,031789131	0,116961708	8,857
	7	0,028456161	0,152157841	14,500
	9	0,004757941	0,024288564	18,430

Т а б л и ц а 8

Результаты эксперимента для меняющегося числа итераций градиентного алгоритма

Тестовая функция	Итерации градиентного алгоритма	Усредненная ошибка		Суммарное время
		абсолютная	максимальная	
1	1000	0,000129491	0,000564288	9,0
	2500	0,000019357	0,000088489	14,5
	5000	0,000001552	0,000007161	24,0
2	1000	0,006566823	0,039670852	25,0
	2500	0,001929114	0,013562487	65,0
	5000	0,001910488	0,013207503	129,0

При настройке параметров нечеткой системы, аппроксимирующей поверхность со многими оптимумами, увеличение числа итераций работы градиентного алгоритма приводит к уменьшению ошибки только в некоторых пределах, определить которые можно лишь экспериментально.

Заключение. Классический АМК, принадлежащий к классу метаэвристик, успешно применяется для решения задач комбинаторной оптимизации. В данной работе предпринята попытка использовать АМК для решения оптимизационной задачи с непрерывно меняющимися параметрами, а именно для решения задачи параметрической идентификации нечетких систем. Переход от одного типа оптимизации к другому выполнен путем построения полного ориентированного графа, количество вершин в котором определяется точностью нахождения значений параметров. Для повышения эффективности и скорости сходимости процедуры идентификации предложено использовать градиентный метод.

Проведенные исследования показали, что гибридный алгоритм, построенный на основе АМК и ГМ, позволяет существенно уменьшить ошибки вывода и время обучения нечеткой системы.

Дальнейшее развитие данной работы предполагает совместное использование различных метаэвристик (метода роящихся частиц, генетических алгоритмов, АМК, локального поиска) для решения задачи идентификации нечетких моделей. Последовательная и независимая схема идентификации себя исчерпала, выбрать какой-либо один эффективный алгоритм решения проблемы оптимизации не представляется возможным ввиду сложности и многоплановости самой проблемы, поэтому необходимо использовать несколько алгоритмов и эвристик одновременно. При параллельном выполнении имеется возможность сотрудничества между эвристиками, а решение, произведенное одной процедурой, может быть впоследствии улучшено другой. Таким образом, можно сократить время выполнения отдельных процедур, улучшить качество решения и повысить устойчивость к ошибкам.

СПИСОК ЛИТЕРАТУРЫ

1. **Serra G., Bottura C.** An IV-QR algorithm for neuro-fuzzy multivariable online identification // IEEE Transaction Fuzzy Systems. 2007. **15**. P. 200.
2. **Chen Y., Yang B., Abraham A., Peng L.** Automatic design of hierarchical Takagi–Sugeno type fuzzy systems using evolutionary algorithms // Ibid. P. 385.
3. **Yu W., Li X.** Fuzzy identification using fuzzy neural network with stable learning algorithms // IEEE Transaction Fuzzy Systems. 2004. **12**. P. 411.
4. **Wu B., Yu X.** Fuzzy modelling and identification with genetic algorithm based learning // Fuzzy Sets and Systems. 2000. **113**. P. 352.
5. **Ходашинский И. А.** Нейросетевой метод обучения нечеткой системы оценивания величин // Нейрокомпьютеры: разработка, применение. 2003. № 12. С. 3.
6. **Ходашинский И. А., Дудин П. А.** Оценивание параметров функций принадлежности на основе алгоритма муравьиной колонии // Тр. Междунар. науч.-техн. конф. «Интеллектуальные системы» (IEEE AIS‘07). М.: Физматлит, 2007. Т. 1. С. 88.
7. **Zadeh L. A.** Fuzzy logic, neural networks, and soft computing // Commun. ACM. 1994. **37**. P. 77.
8. **Dorigo M., Maniezzo V., Colorni A.** Ant system: optimization by colony of cooperating agents // IEEE Transaction Systems, Man and Cybernetics. Pt. B. 1996. **26**. P. 29.
9. **Ходашинский И. А.** Оценивание величин: подход на основе мягких вычислений // Информационные технологии. 2006. № 4. С. 13.

Поступила в редакцию 12 ноября 2007 г.